New trends, technologies and tools in Modeling and Simulation

Dr. David A. Cook

Department of Computer Science

Stephen F. Austin State University

Nacogdoches, TX 75962

cookda@sfasu.edu

		Form Approved
Report Documentation Page		OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **18 MAY 2011**	2. REPORT TYPE	3. DATES COVERED **00-00-2011 to 00-00-2011**
4. TITLE AND SUBTITLE **New trends, technologies and tools in Modeling and Simulation**		5a. CONTRACT NUMBER
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)		5d. PROJECT NUMBER
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Stephen F. Austin State University,Department of Computer Science,Nacogdoches,TX,75962**		8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
Presented at the 23rd Systems and Software Technology Conference (SSTC), 16-19 May 2011, Salt Lake City, UT. Sponsored in part by the USAF. U.S. Government or Federal Rights License

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**31**	

What is Simulation?

- *Simulation* – very broad term – methods and applications to imitate or mimic real systems, usually via computer

- Applies in many fields, industries

- Very popular, powerful way to save time and money.

- When used effectively, speeds up "model to reality" by allowing visualization and validity testing of the model

Where is simulation used?

- Manufacturing facility
- Banks
- Airport operations (passengers, security, crews, baggage)
- Transportation/logistics/distribution operation
- Hospital facilities (ERs, operating room, admissions)
- Computer networks
- Freeways
- Medical and Surgical Training
- Fast-food restaurants, supermarkets
- Theme park
- Emergency-response system
- Shipping ports, berths
- Military combat, logistics

Why use simulation?

- Study system – measure, improve, design, control
  - Maybe just investigate changes to actual system
    - Advantage — unquestionably looking at the right thing
  - But often impossible in reality with actual system
    - System doesn't exist
    - Would be disruptive, expensive, dangerous
    - Would result in loss of lives

Using Computers to Simulate

- General-purpose languages (C, C++, C#, Java, Matlab, FORTRAN, others)
  - Tedious, low-level, error-prone
  - But, almost complete flexibility
- Support packages for general-purpose languages
  - Subroutines for list processing, bookkeeping, time advance
  - Widely distributed, widely modified
- Spreadsheets
  - Usually static models (only *very* simple dynamic models)
  - Financial scenarios, distribution sampling, SQC
  - Examples in Chapter 2 (one static, one dynamic)
  - Add-ins are available (@RISK, Crystal Ball)

Using Computers to Simulate (cont'd.)

- Simulation languages
  - GPSS, SLX,
  - Popular, some still in use
  - Learning curve for features, effective use, syntax
- High-level simulators
  - Very easy, graphical interface
  - Domain-restricted (manufacturing, communications)
  - Limited flexibility — need to make sure model is valid

When Simulations are Used

- Use of simulation has evolved with hardware, software

- Early years (1950s – 1960s)
  - Very expensive, specialized tool
  - Required big computers, special training
  - Mostly in FORTRAN (or even Assembler)
  - Processing cost as high as $1000/hour for a sub-PC level machine

When Simulations are Used (cont'd.)

- Formative years (1970s – early 1980s)
  - Computers got faster, cheaper
  - Value of simulation more widely recognized
  - Simulation software improved, but still languages to be learned, typed, batch processed
  - Often used to clean up "disasters" in auto, aerospace industries
    - Car plant; heavy demand for certain model
    - Line underperforming
    - Simulated, problem identified
    - But demand had dried up — simulation was too late

When Simulations are Used (cont'd.)

- Recent past (late 1980s – mid 2000s)
  - Microcomputer power
  - Software expanded into GUIs, animation
  - Wider acceptance across more areas
    - Traditional manufacturing applications
    - Services
    - Health care
    - "Business processes"
  - Still mostly in large firms
  - Simulation is often part of "specs"

The Search:: 1955 - 60

- 1955 – 1960: Fortran was King, General Simulation Program was envisioned. Fortran based (with reusable functions)

- 1960s: GPSS (IBM, queueing models). Also SIMSCRIPT and SIMSCRIPT II (Rand Corp. and USAF). GASP (Algol and Fortran), and SIMULA (mostly Europe.

- 1970s – GPSS/H, GASP IV, SIMULA
  - Attempt to simplify the modeling process
  - Program generators – severe limitations

Next Leap Forward – the 1980s

- Movement to mini and PC computers
- SLAM II (descendant of GASP)
  - 3 world views
    - Event, Network, Continuous
- SIMAN (descendant of GASP)
  - General Modeling + Block Diagrams
  - 1st first major language - PC & MS-DOS
  - Fortran functions w/ Fortran programming

1980s – Present Integrated Environments

- Growth on PC's

- Simulation Environments
  - GUI
  - Animation
  - Data analyzers

The Future (NOW)

- Virtual Reality
- Improved Interfaces
- Better Animation
- Agent-based Modeling
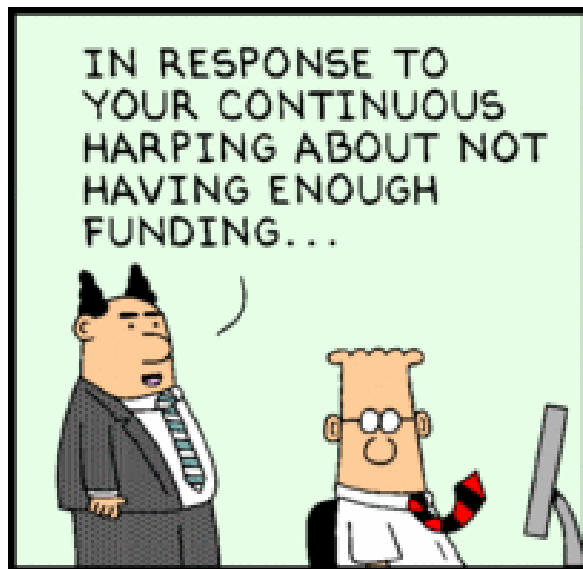
How do you know the model is correct?
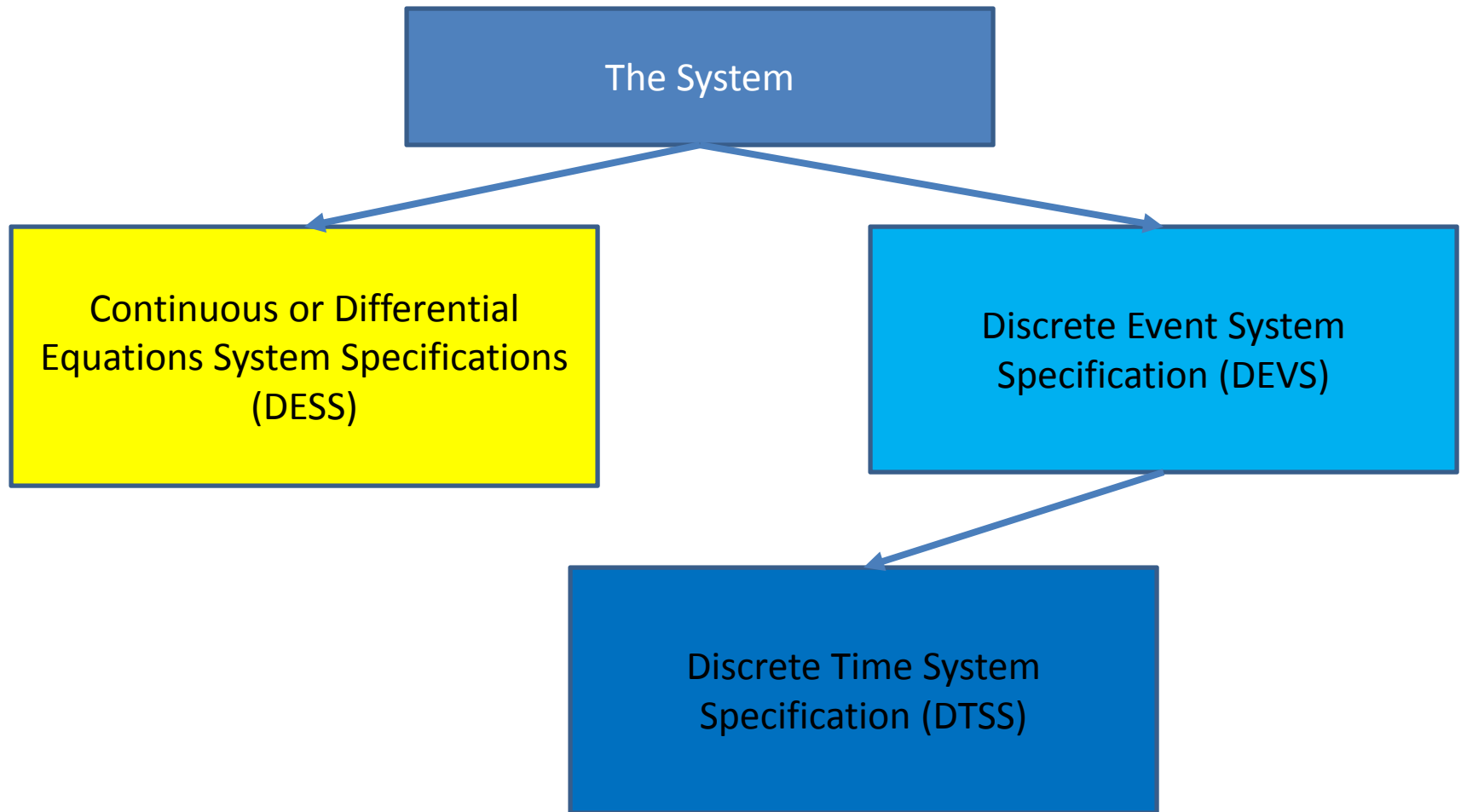
- Simulation Validity
  - Structural Simulation
  - Behavioral Simulation
  - Predictive Simulation

- Simulation Verification
  - How the simulation is built
  - What estimates are used
  - What are the unknowns
  - How accurate and of what fidelity are the results

Types of Simulation

The System

Continuous or Differential Equations System Specifications (DESS)

Discrete Event System Specification (DEVS)

Discrete Time System Specification (DTSS)

Dynamic Simulations

- Can be
  - Continuous – changes constantly over time. For any exact time (i.e. t = 1 hour, 2 minutes, 13.9987665 seconds) there is a (potentially) exact value
  - Discrete – changes occur at specific and separated points in time. For example, a customer can arrive at a bank at 3:14:15, but not again until 3:14:16. It is "impossible" (i.e. a non-event) for a customer to arrive halfway between two time steps.

- Continuous and Dynamic simulations (mixed models) are possible

Discrete Event Simulations

- Discrete – changes occur at specific and separated points in time.

- For example, a customer can arrive at a bank at 3:14:15, but not again until 3:14:16. It is "impossible" (i.e. a non-event) for a customer to arrive halfway between two time steps.

Deterministic vs. Stochastic

- Deterministic – there is no element of randomness in the simulation.

- Stochastic – some part of the simulation is based on randomness. Randomness can be event-based (customer entrance time) or probability-based (the odds of an event, like a structural failure) occurring.

And the difference...

- DESS – specific time $dq/dt = a*x + b$ for all t
  - Every time t has a specific value, not necessarily dependent upon any other time t

- DTSS – $q(t + 1) =$ some function of $q(t)$
  - Every time has value (state) based on previous time

- DEVS – there is a time t_n of the next event.
  - The state at the next event is a function of all events that have preceded the event.

All good simulations based on a model

- A simulation must be designed to either

  - Model a real system. The system can them be used for comparisons and verification and validation

  - Model an imaginary system (that might or might not be build in the future). Verification and validation much harder.

How do you know the model is correct?

- Validity
  - Structural Simulation
  - Behavioral Simulation
  - Predictive Simulation

- Verification
  - How the simulation is built
  - What estimates are used
  - What are the unknowns
  - How accurate and of what fidelity are the results

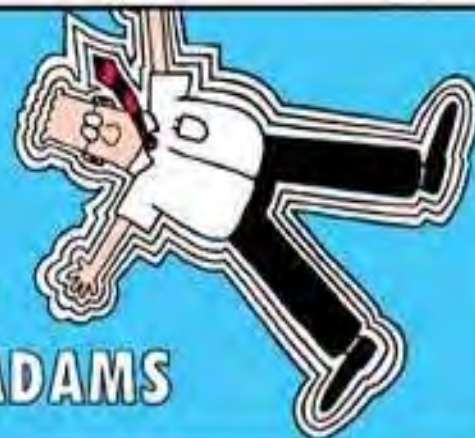# What's new in Simulation today?

- Graphical simulation languages can be used to create the model, run the simulation, and explore the outputs.

- GUIs of varying levels of detail and specificity can be used to build complex graphical models

How to build a M&S

- Hierarchical structure
  - Multiple levels of modeling
  - Mix different modeling levels together in same model
  - Often, start high then go lower as needed
- Get ease-of-use advantage of simulators without sacrificing modeling flexibility

Higher

User-Created Templates
Commonly used constructs
Company-specific processes
Company-specific templates
etc.
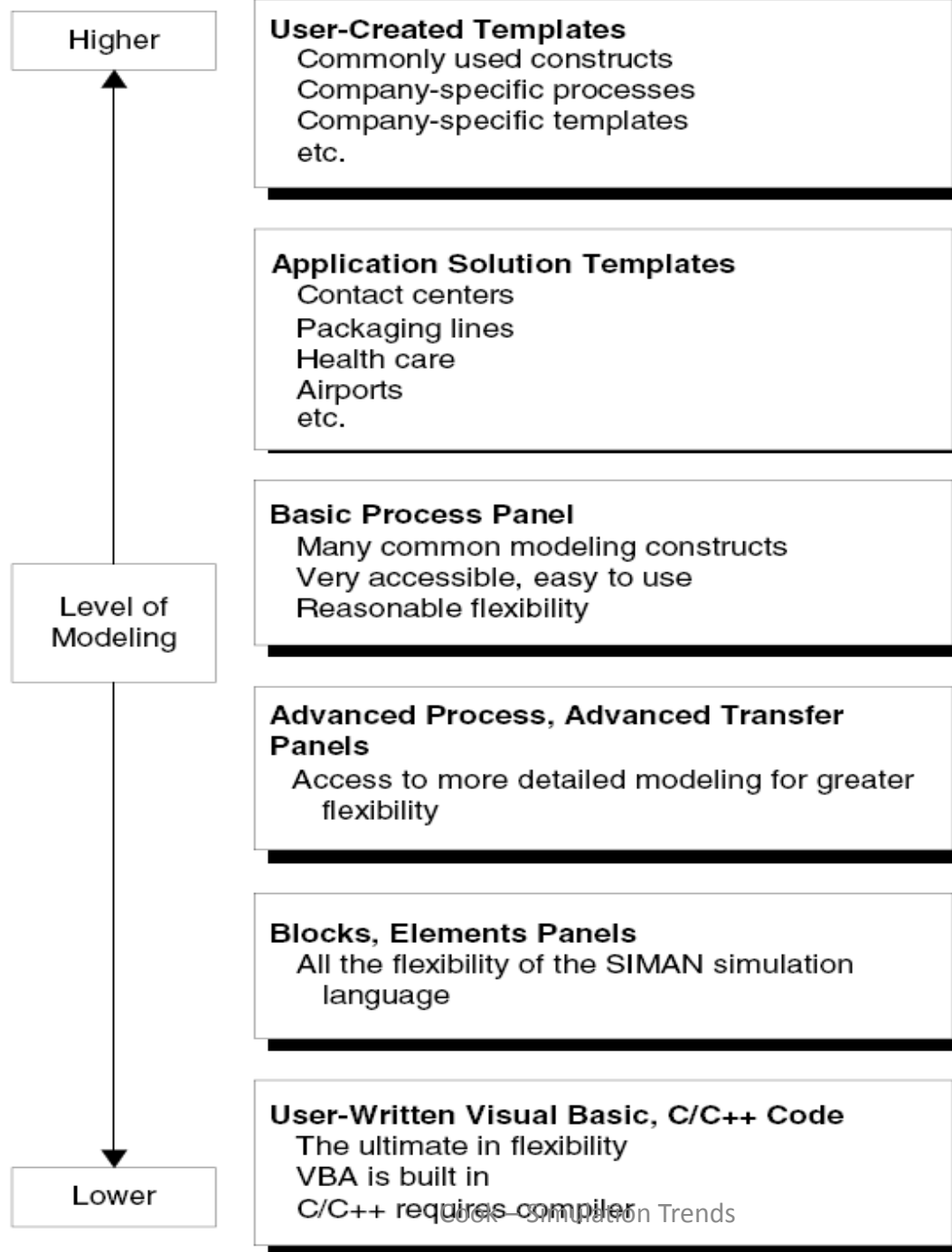
A single
graphical user
interface
consistent at
any level of
modeling

Application Solution Templates
Contact centers
Packaging lines
Health care
Airports
etc.

Basic Process Panel
Many common modeling constructs
Very accessible, easy to use
Reasonable flexibility

Level of
Modeling

Advanced Process, Advanced Transfer Panels
Access to more detailed modeling for greater
flexibility

Blocks, Elements Panels
All the flexibility of the SIMAN simulation
language

User-Written Visual Basic, C/C++ Code
The ultimate in flexibility
VBA is built in
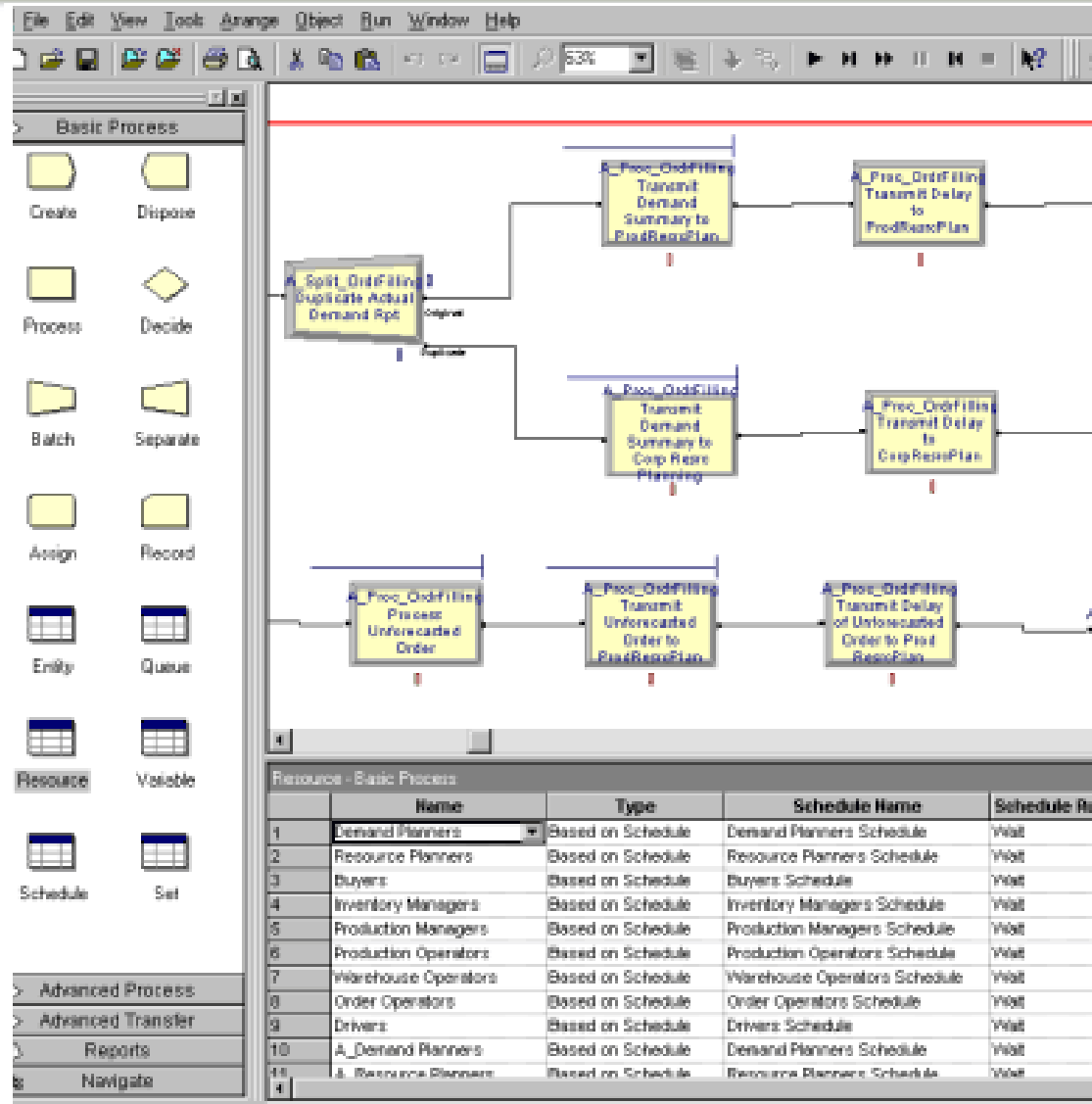C/C++ requires compiler

Lower

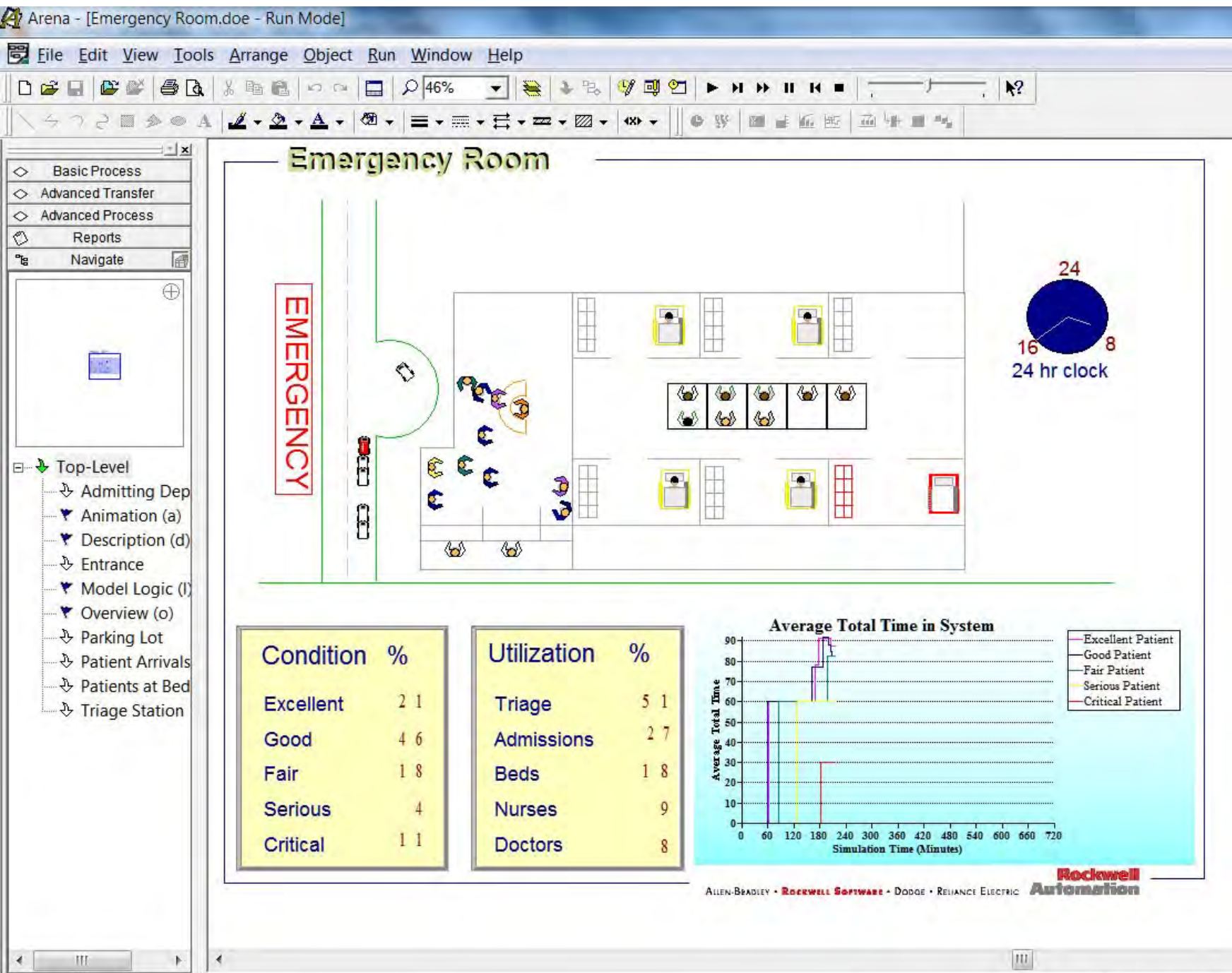# Visual and Graphical M&S Trends

- **Highly realistic training scenarios**

- **Training realistic than games designed for entertainment**

- **Focus on end-user experience that matches real life (virtual reality)**

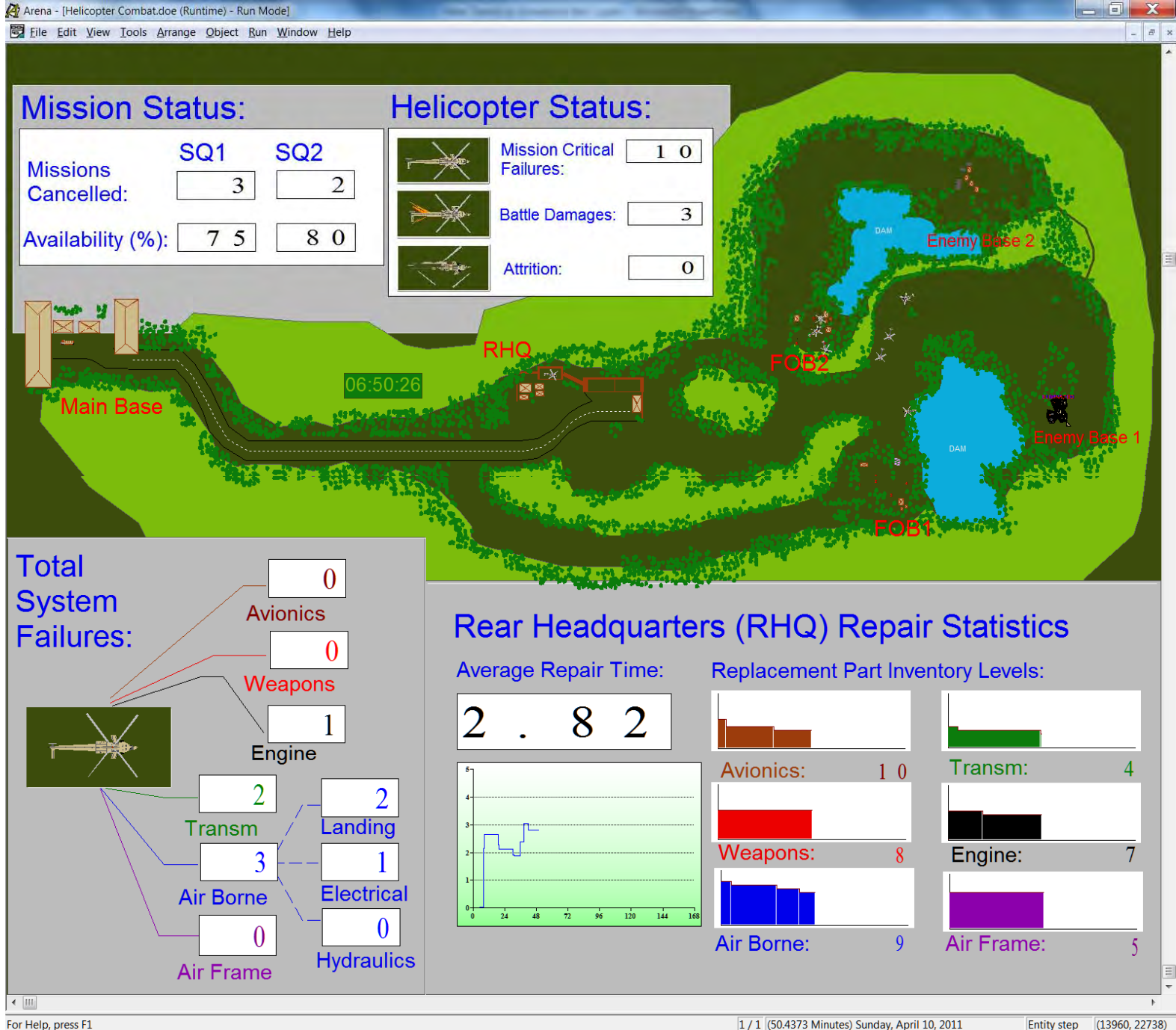# However = lots of $$$s NOT needed!

- Simple languages and available tools let you transform models and simulations into easy to visualize products that can be used for
    - Proof of concept
    - Ease of user understanding
    - Graphical display of formerly table-driven data

- This speeds up the "model to reality" timeline

One Example - Arena

Cook – Simulation Trends

Mission Status:

Helicopter Status:

	SQ1	SQ2
Missions Cancelled:	3	2
Availability (%):	7 5	8 0

Mission Critical Failures: 1 0

Battle Damages: 3

Attrition: 0

Main Base

06:50:26

RHQ

DAM

Enemy Base 2

FOB2

Enemy Base 1

DAM

FOB1

Total System Failures:

Avionics 0

Weapons 0

Engine 1

Transm 2

Landing 2

Air Borne 3

Electrical 1

Air Frame 0

Hydraulics 0

Rear Headquarters (RHQ) Repair Statistics

Average Repair Time:

2 . 8 2

Replacement Part Inventory Levels:

Avionics: 1 0

Transm: 4

Weapons: 8

Engine: 7

Air Borne: 9

Air Frame: 5

Questions or comments??

Dr. David A. Cook

Department of Computer Science

Stephen F. Austin State University

Nacogdoches, TX 75962

cookda@sfasu.edu